Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer-implemented method for identifying ~~one or more~~ multiple objects within an image, the method comprising:

receiving, in a programmable computer system, an image that includes two or more non-overlapping embedded images from an input device coupled to the programmable computer system;

identifying a plurality of edge pixels in the image based on a respective gradient value associated with each of the plurality of edge pixels, wherein the gradient value of each edge pixel satisfies a first threshold value;

identifying one or more non-edge pixels in the image based on a respective gradient value associated with each of the plurality of non-edge pixels, wherein the gradient value of each non-edge pixel satisfies a second threshold value;

selecting an edge pixel from the plurality of edge pixels;

identifying a substantially connected component that includes non-edge pixels and a plurality of substantially connected edge pixels being substantially connected to the selected edge pixel, wherein ~~the~~ a number of non-edge pixels in the substantially connected component is based on a level of tolerance for non-edge pixels; and

~~establishing~~ identifying within the image a bounding area, wherein the bounding area ~~comprises a perimeter that is different from the substantially connected component and~~ surrounds the substantially connected component.

2. (Canceled)

3. (Previously Presented) The method of claim 1, wherein identifying a plurality of edge pixels includes computing the respective gradient value for each of a plurality of pixels in the image.

4. (Previously Presented) The method of claim 3, wherein computing the respective gradient value for each of the plurality of pixels includes, for each pixel, comparing respective pixel colors of a neighborhood of pixels surrounding each pixel.

5. (Previously Presented) The method of claim 3, wherein computing the respective gradient value for each of the plurality of pixels includes using an image smoothing filter to filter noise from the image.

6. (Previously Presented) The method of claim 1, further comprising passing the bounding area to a processor that extracts a location of an embedded image from the two or more non-overlapping images.

7. (Previously Presented) The method of claim 6, further comprising refining the extracted location.

8. (Previously Presented) The method of claim 7, further comprising using the extracted location to crop the embedded image from the image.

9. (Previously Presented) The method of claim 1, further comprising splitting the bounding area of the image into a first of the two or more non-overlapping embedded images and a second of the two or more non-overlapping embedded images.

10. (Previously Presented) The method of claim 1, further comprising merging the bounding area within the image with another bounding area within the image into a single bounding area.

11. (Previously Presented) The method of claim 1, further comprising:

extracting a location of each of the two or more non-overlapping embedded images from the image; and

using the location to seed a crop operation.

12. (Previously Presented) The method of claim 11, wherein using the location to seed a crop operation includes:

for each of the two or more non-overlapping embedded images in the image, using the location to define a cropping area; and

cropping all the defined cropping areas in a single cropping operation.

13. (Previously Presented) The method of claim 12, wherein:

the location specifies an alignment of one of the two or more non overlapping embedded images with respect to the image; and

using the location to define a cropping area includes using the alignment of one of the two or more non overlapping embedded images to define an alignment of the cropping area.

14. (Previously Presented) The method of claim 12, further comprising:

prior to cropping all the defined cropping areas, adjusting one or more of the defined cropping areas in response to user input.

15. (Previously Presented) The method of claim 12, further comprising:

prior to cropping all the defined cropping areas, merging two of the defined cropping areas into a single defined cropping area in response to user input.

16. (Previously Presented) The method of claim 14, wherein adjusting one or more of the defined cropping areas includes splitting a single cropping area into two or more cropping areas.

17. (Currently Amended) A computer program product, tangibly stored on a computer-readable medium, for identifying ~~one or more~~ multiple objects within an image, comprising instructions operable to cause a programmable processor to perform operations comprising:

receiving an image that includes two or more non-overlapping embedded images;

identifying a plurality of edge pixels in the image based on a respective gradient value associated with each of the plurality of edge pixels, wherein the gradient value of each edge pixel satisfies a first threshold value;

identifying one or more non-edge pixels in the image based on a respective gradient value associated with each of the plurality of non-edge pixels, wherein the gradient value of each non-edge pixel satisfies a second threshold value;

selecting an edge pixel from the plurality of edge pixels;

identifying a substantially connected component that includes non-edge pixels and a plurality of substantially connected edge pixels being substantially connected to the selected edge pixel, wherein a number of non-edge pixels in the substantially connected component is based on a level of tolerance for non-edge pixels; and

~~establishing~~ identifying within the image a bounding area, wherein the bounding area ~~comprises a perimeter that is different from the substantially connected component and~~ surrounds the substantially connected component.

18. (Canceled)

19. (Previously Presented) The product of claim 17, wherein identifying a plurality of edge pixels includes computing the respective gradient value for each of a plurality of pixels in the image.

20. (Previously Presented) The product of claim 19, wherein computing the respective gradient value for each of the plurality of pixels includes, for each pixel, comparing respective pixel colors of a neighborhood of pixels surrounding each pixel.

21. (Previously Presented) The product of claim 19, wherein computing the respective gradient value for each of the plurality of pixels includes using an image smoothing filter to filter noise from the image.

22. (Previously Presented) The product of claim 17, wherein the operations further comprise passing the bounding area to a processor that extracts a location of an embedded image from the two or more non-overlapping images.

23. (Previously Presented) The product of claim 22, wherein the operations further comprise refining the extracted location.

24. (Previously Presented) The product of claim 23, wherein the operations further comprise using the extracted location to crop the embedded image from the image.

25. (Previously Presented) The product of claim 17, wherein the operations further comprise splitting the bounding area of the image into a first of the two or more non-overlapping embedded images and a second of the two or more non-overlapping embedded images.

26. (Previously Presented) The product of claim 17, wherein the operations further comprise merging the bounding area within the image with another bounding area within the image into a single bounding area.

27. (Previously Presented) The product of claim 17, wherein the operations further comprise:

extracting a location of each of the two or more non-overlapping embedded images from the image; and

using the location to seed a crop operation.

28. (Previously Presented) The product of claim 27, wherein using the location to seed a crop operation includes:

for each of the two or more non-overlapping embedded images in the image, using the

location to define a cropping area; and

cropping all the defined cropping areas in a single cropping operation.

29. (Previously Presented) The product of claim 28, wherein:

the location specifies an alignment of one of the two or more non overlapping embedded images with respect to the image; and

using the location to define a cropping area includes using the alignment of one of the two or more non overlapping embedded images to define an alignment of the cropping area.

30. (Previously Presented) The product of claim 28, wherein the operations further comprise:

prior to cropping all the defined cropping areas, adjusting one or more of the defined cropping areas in response to user input.

31. (Previously Presented) The product of claim 28, wherein the operations further comprise:

prior to cropping all the defined cropping areas, merging two of the defined cropping areas into a single defined cropping area in response to user input.

32. (Previously Presented) The product of claim 30, wherein adjusting one or more of the defined cropping areas includes splitting a single cropping area into two or more cropping areas.

33. (Currently Amended) A computer program product, tangibly stored on a computer-readable medium, for identifying multiple objects within a scanned image, comprising instructions operable to cause a programmable processor to perform operations comprising:

receiving a scanned image that includes multiple non-overlapping embedded images; and

identifying the multiple non-overlapping embedded images by:

(1) generating an edge pixel map of the image based on a respective gradient value associated with each pixel in the scanned image, the edge pixel map identifying each pixel

in the scanned image as being an edge pixel or a non-edge pixel, an edge pixel being a pixel having a gradient value satisfying a first threshold value and a non-edge pixel being a pixel satisfying a second threshold value;

(2) scanning the edge pixel map until an edge pixel is found;

(3) identifying a substantially connected component containing the edge pixel, the substantially connected component being a set of edge pixels that are substantially connected by traversing adjacent edge pixels and adjacent non-edge pixels, wherein a number of non-edge pixels in the substantially connected component is based on a level of tolerance for non-edge pixels;

(4) ~~establishing~~ identifying within the image a bounding area ~~different from the substantially connected component, the bounding area comprising a perimeter that is different from the substantially connected component and surrounds~~ surrounding the substantially connected component, and extracting one of the multiple non-overlapping embedded images based on the bounding area;

(5) erasing from the edge pixel map all the pixels that belong to the substantially connected component or that correspond to pixels inside the extracted one of the multiple non-overlapping embedded images; and

(6) repeating steps (2) through (5) until no more edge pixels are found.

34. (Previously Presented) The method of claim 1 further comprising setting the tolerance level based on a user input.

35. (Previously Presented) The method of claim 1 further comprising automatically determining the tolerance level as a function of a spacing between the objects.

36. (Previously Presented) The product of claim 17, further comprising instructions operable to cause the programmable processor to set the tolerance level based on a user input.

37. (Previously Presented) The product of claim 17, further comprising instructions operable to cause the programmable processor to automatically determine the tolerance level as a function of a spacing between the embedded images.

38. (Previously Presented) The product of claim 33, further comprising instructions operable to cause the programmable processor to set the tolerance level based on a user input.

39. (Previously Presented) The product of claim 33, further comprising instructions operable to cause the programmable processor to automatically determine the tolerance level as a function of a spacing between the embedded images.

40. (Currently Amended) A system comprising:

a display device;

a machine-readable storage device including a program product; and

one or more processors operable to execute the program product, interact with the display device, and perform operations comprising:

receiving an image that includes two or more non-overlapping embedded images;

identifying a plurality of edge pixels in the image based on a respective gradient value associated with each of the plurality of edge pixels, wherein the gradient value of each edge pixel satisfies a first threshold value;

identifying one or more non-edge pixels in the image based on a respective gradient value associated with each of the plurality of non-edge pixels, wherein the gradient value of each non-edge pixel satisfies a second threshold value;

selecting an edge pixel from the plurality of edge pixels;

identifying a substantially connected component that includes non-edge pixels and a plurality of substantially connected edge pixels being substantially connected to the selected edge pixel, wherein a number of non-edge pixels in the substantially connected component is based on a level of tolerance for non-edge pixels; and

~~establishing~~ identifying within the image a bounding area, wherein the bounding

area ~~comprises a perimeter that is different from the substantially connected component and~~ surrounds the substantially connected component.

41. (Previously Presented) The system of claim 40 wherein identifying a plurality of edge pixels includes computing the respective gradient value for each of a plurality of pixels in the image.

42. (Previously Presented) The system of claim 41 wherein computing the respective gradient value for each of the plurality of pixels includes, for each pixel, comparing respective pixel colors of a neighborhood of pixels surrounding each pixel.

43. (Previously Presented) The system of claim 41 wherein computing the respective gradient value for each of the plurality of pixels includes using an image smoothing filter to filter noise from the image.

44. (Previously Presented) The system of claim 40 wherein the one or more processors are operable to perform operations further comprising extracting a location of an embedded image from the two or more non-overlapping images .

45. (Previously Presented) The system of claim 44 wherein the one or more processors are operable to perform operations further comprising refining the extracted location.

46. (Previously Presented) The system of claim 45 wherein the one or more processors are operable to perform operations further comprising using the extracted location to crop the embedded image from the image.

47. (Previously Presented) The system of claim 40 wherein the one or more processors are operable to perform operations further comprising splitting the bounding area of the image into a first of the two or more non-overlapping embedded images and a second of the two or more non-overlapping embedded images.

48. (Previously Presented) The system of claim 40 wherein the one or more processors are operable to perform operations further comprising merging the bounding area within the image with another bounding area within the image into a single bounding area.

49. (Previously Presented) The system of claim 40 wherein the one or more processors are operable to perform operations further comprising:

extracting a location of each of the two or more non-overlapping embedded images from the image; and

using the location to seed a crop operation.

50. (Previously Presented) The system of claim 49 wherein using the location to seed a crop operation includes:

for each of the two or more non-overlapping embedded images in the image, using the location to define a cropping area; and

cropping all the defined cropping areas in a single cropping operation.

51. (Previously Presented) The system of claim 50 wherein:

the location specifies an alignment of one of the two or more non overlapping embedded images with respect to the image; and

using the location to define a cropping area includes using the alignment of one of the two or more non overlapping embedded images to define an alignment of the cropping area.

52. (Previously Presented) The system of claim 50 wherein the one or more processors are operable to perform operations further comprising:

prior to cropping all the defined cropping areas, adjusting one or more of the defined cropping areas in response to user input.

53. (Previously Presented) The system of claim 50 wherein the one or more processors are operable to perform operations further comprising:

prior to cropping all the defined cropping areas, merging two of the defined cropping areas into a single defined cropping area in response to user input.

54. (Previously Presented) The system of claim 52 wherein adjusting one or more of the defined cropping areas includes splitting a single cropping area into two or more cropping areas.